

VITAL: More Understandable Feature Visualization through Distribution Alignment and Relevant Information Flow

Supplementary Material

1. Method

Here, we provide detailed information about the VITAL framework, the sort-matching procedure, as well as further implementation considerations for all tested methods. In Fig. 1, we give an overview of VITAL, with an example computation of the sort matching (SM) algorithm and its corresponding pseudo-code provided in Alg. 1.

1.1. Implementation Details

For the experiments, we use a publicly available pretrained models (ResNet50 [8], DenseNet121 [9], ConvNeXt-base [11], ViT-L-16 [4], ViT-L-32 [4]) from the PyTorch [13] library. We report feature visualizations (FVs) of all methods across three different random seeds for each category of the ImageNet dataset [3]. In detail,

- for **VITAL**, we synthesize a single image with resolution 224×224 and apply jittering at each optimization step to promote robustness. We set the number of real images in our reference dataset \mathcal{X}_{ref} for the feature distribution matching as $N = 50$. For the polysemanticity experiments further below, we similarly set $N = 50$ and consider 1000 patches for k -Means. For optimizing the feature visualization, we use Adam with a learning rate of 1.0. For intermediate neuron visualizations, we select the patch size as 64 and set the scales $\alpha_{TV} = \alpha_{\ell_2} = 3 \times 10^{-6}$, $\lambda = 1$. We provide ablations for the effects of $\alpha_{TV}, \alpha_{\ell_2}$ in Sec. 3.2. After analyzing the effect of each network component of ResNet50 on our SM loss in Sec. 3.1, we decided to utilize all the network components. Specifically, For ResNet50, for class neurons, the loss weight for each block is set to 1.0 whereas for intermediate neurons, we reduce the contribution of block1 to be 0.1. For DenseNet121 with class neurons, the loss weight for each block is set to 1.0 except the final block, which is set to 100.0. For ConvNeXt-base with class neurons, the loss weight for each block is set to 1.0 except the first block, which is set to 10.0. For ViT-L-16 with class neurons, for the selected 5 blocks that includes the projection layer and selected encoder layers, the loss weight for each block is set to 1.0. Finally, for ViT-L-32 with class neurons, for the selected 5 blocks that includes the projection layer and selected encoder layers, the loss weight for each block is set to 1.0 except the final block, which is set to 0.1. For the experiments involving the visualization of class neurons using LRP with ResNet50, we additionally utilized auxiliary regularization with parameters $\alpha_{TV} = \alpha_{\ell_2} = 0.00001$.

- for **DeepInversion** [16], we synthesize a batch of images with resolution 224×224 and apply jittering at each optimization step to promote robustness. We adapted the parameters from their official GitHub implementation [2]. In detail, we use Adam for optimization with a learning rate of 0.05, and set the scales of the auxiliary regularization as $\alpha_{TV} = 0.0001$, $\alpha_{\ell_2} = 0.00001$, $\lambda = 1$. For a fair comparison, we do not apply the teacher-student guidance that was proposed in Adaptive DeepInversion.
- for **MACO** [5], we synthesize both regular (224×224) and high resolution (1024×1024) images offered by MACO and we found that higher resolution visualizations were more human readable. Yet, as shown in the quantitative experiments, this effect seemed more like a subjective, qualitative finding and did not carry over to CLIP Zero-shot prediction scores, classification scores, or FID scores. As suggested by Fel et al. [5], for transformations, we first add uniform noise $\delta \sim \mathcal{U}([-0.1, 0.1])^{W \times H}$ and augment the data at each iteration with crops of the input image that are resized to (224×224), in which the crop size drawn from the normal distribution $\mathcal{N}(0.25, 0.1)$. For optimization, we use the Adam optimizer with a learning rate of 1.0.
- for **Fourier** [12], we use the same settings as for MACO, only the initialization of the generated image is changed to regular Fourier initialization, i.e., without fixed magnitude.
- for **PfII** [7], we use the published implementation, including the provided batch-size settings for all models except ConvNext-base (not implemented) with image resolution of 224×224 .

Algorithm 1 SM Loss for layer- l

Input: $f_l(x) \subseteq \mathbb{R}^{1 \times C \times HW}$, $f_l(y) \subseteq \mathbb{R}^{N \times C \times HW}$
 $-$, IndexX = torch.sort($f_l(x)$, dim = 2)
SortedY, $-$ = torch.sort($f_l(y)$, dim = 2)
SortedY = torch.mean(SortedY, dim = 0)
InverseIndex = IndexX.argsort(-1)
 $g_l(y) = \text{SortedY.gather}(-1, \text{InverseIndex})$
return $\mathcal{L}_{MSE} = \text{torch.mean}((f_l(x) - g_l(y))^2)$

1.2. CLIP Zero-shot Prediction

To evaluate the FVs based on how "understandable" they are in terms of the target class they aim to visualize, we considered a pretrained CLIP model. The CLIP space [14] is an effective method for quantifying visualization methods

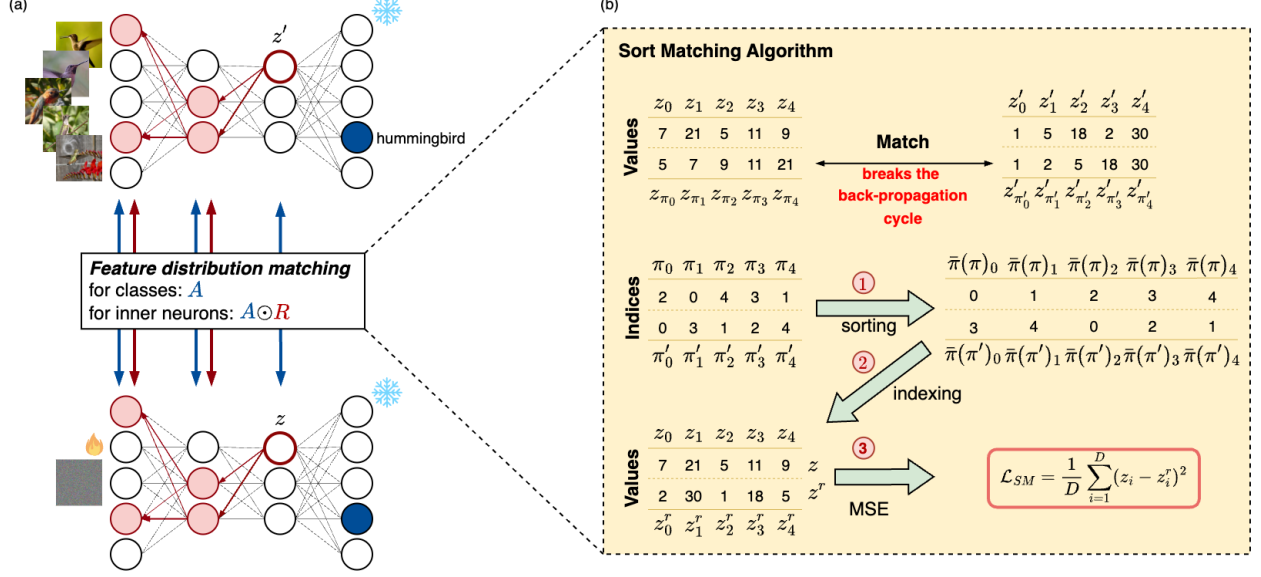


Figure 1. *Overview of VITAL framework.* Our VITAL framework mainly consists of two main components. In part (a), we utilize a pretrained *frozen* model (*) to generate visualizations. For the given type of visualization-class neurons or intermediate neurons—we first select N reference images from the ImageNet training dataset, then we optimize a randomly initialized image (🔥) through our feature distribution matching approach, which is applied across multiple layers of the model. For feature distribution matching, we utilize the feature activations A and feature relevance (arrows) $A \odot R$ for class neurons and intermediate neurons, respectively. We perform the feature distribution matching using (b) our sort matching loss, where we compute the difference between the feature distributions of z (from synthetic data) and z' (from reference data). We achieve this by first sorting the values and obtaining the sorted indices as π and π' . Considering that sorting is a discrete operation and we want to allow back-propagation to optimize z , we obtain a reverse mapping $\bar{\pi}(\cdot)$ by ① sorting the indices π' to ② re-index z' to z^r , which allows z to be unchanged. Thus, we were able to ③ match z and z^r through minimizing the MSE loss. The proposed SM loss can be used in a plug-and-play manner without introducing any parameters, as summarized in Alg. 1.

as it bridges the gap between image and text, thus offering a powerful measure of how well a visualization aligns with an intended concept. We perform this experiment to understand how a *different* model *perceives* the visual features presented in a FV. Specifically, we load a pretrained CLIP ViT-B/32 model and expand the ImageNet labels into descriptive textual prompts (80 templates) [1], such as "a photo of a {label}" or "a picture of a {label}". The main goal is to explore how different descriptions of the same class affect the model's predictions by varying the phrasing of these prompts. We divide the calculation of the zero-shot prediction scores into two steps. In the first step, we first associate each class label with the textual prompts and for each prompt, we compute its embedding using CLIP's text encoder. These embeddings are then averaged to obtain a single representative embedding for each class, which we refer to as their corresponding "zero-shot weight". Each class is hence represented by a robust and generalized text embedding. In the second step, we compute the embeddings of the input images, including our FVs, into CLIP's shared feature space using its respective encoders. Then, we compute the cosine similarities between the zero-shot weights of each class and the image embeddings to identify the best-

matching class for each image. The performance is evaluated by measuring the zero-shot classification accuracy on the original (correctly classified) ImageNet images as well as the images of different FV methods.

1.3. Selection of Reference Images

We define our image set as the entire ImageNet training data. Considering the computational needs, a subset of the training data could also be used. After defining our image set, we divide our selection of \mathcal{X}_{ref} for the two sub-problems, (1) class visualizations and (2) intermediate neuron visualizations. For (1), we select N random images for feature distribution matching. In (2), we first select sub-regions to identify specific concepts that neurons respond to within a localized context. Similar to CRAFT [6], we first crop and resize the training images into patches to obtain an auxiliary dataset. Then, we obtain the activations for each patch, keeping the top- N patches while eliminating the patches that are coming from the same original image. In the case where the model is a CNN, the score for each patch is formulated as the global average pooling of activations across their spatial dimension.

	Method	Setup	Acc.	FID (↓)		Zero-Shot Prediction	
			Top1 (↑)	RN50	Arch.	Top1 (↑)	Top5 (↑)
ResNet50	ImageNet		-	-	-	69.11	92.23
	MACO	r: 224	29.43	360.74	360.74	12.87	29.73
		r: 1024	2.10	494.57	494.57	1.60	5.67
	Fourier	r: 224	21.30	422.44	422.44	6.73	18.27
		r: 1024	3.43	430.58	430.58	0.97	3.57
	DeepInversion	bs: 64	100.00	35.76	35.76	29.90	55.20
		bs: 1	100.00	123.77	123.77	4.73	12.63
	DeepInversion (↓ 2)	bs: 64	50.47	176.35	176.35	17.20	40.43
		bs: 1	100.00	121.94	121.94	6.30	16.43
	PII	bs: 21	100.00	241.54	241.54	17.53	38.93
VITAL	train-set	<u>99.90</u>	<u>58.79</u>	<u>58.79</u>	66.62	92.56	
ConvNeXt base	ImageNet		-	-	-	65.66	89.80
	MACO	r: 224	66.07	<u>369.64</u>	62.55	<u>7.20</u>	<u>19.77</u>
		r: 1024	21.07	495.69	97.73	1.07	4.73
	Fourier	r: 224	60.07	453.91	<u>59.60</u>	2.77	8.30
		r: 1024	14.27	529.33	77.08	0.60	2.37
	PII	bs: 16	100.00	405.50	92.37	1.97	6.47
	VITAL		<u>99.97</u>	88.63	3.92	63.53	90.30
DenseNet121	ImageNet		-	-	-	70.64	93.16
	MACO	r: 224	9.20	418.60	1.80	9.33	23.20
		r: 1024	1.60	475.39	1.98	1.43	5.03
	Fourier	r: 224	15.53	409.89	1.63	4.87	12.17
		r: 1024	1.80	437.18	1.88	0.90	3.33
	DeepInversion	bs: 64	100.00	<u>93.26</u>	0.20	10.00	<u>25.47</u>
	DeepInversion (↓ 2)	bs: 64	31.30	186.16	0.83	7.23	20.03
	PII	bs: 24	100.00	377.92	1.23	<u>11.00</u>	24.00
VITAL		<u>99.93</u>	79.40	<u>0.27</u>	58.70	86.93	
ViT-L-16	ImageNet		-	-	-	64.78	89.31
	MACO	r: 224	44.33	371.54	946.96	3.93	10.57
		r: 224	25.30	447.56	990.51	1.67	5.13
	PII	bs: 2	100.00	274.28	<u>537.32</u>	15.70	32.73
	VITAL		<u>99.80</u>	128.02	126.29	68.17	92.80
ViT-L-32	ImageNet		-	-	-	65.83	90.03
	MACO	r: 224	24.87	280.77	2318.90	17.53	37.23
		r: 224	17.03	331.86	1983.09	10.30	28.10
	PII	bs: 5	100.00	<u>270.20</u>	<u>293.02</u>	38.47	67.40
	VITAL		<u>89.60</u>	174.31	147.33	55.97	85.47

Table 1. Comparison of methods on different architectures trained on Imagenet. We provide FID scores, CLIP Zero-shot prediction scores, and top-1 classification accuracy, indicating the **best** and **second best**. In the settings, "r" indicates the resolution of the visualization, "bs" is the used batch size and indicate with (↓ 2) the multi-resolution optimization version of DeepInversion.

2. Additional Results

In this section, we provide more quantitative results (Tab. 1, with various setups for methods Fourier [12], MACO [5], DeepInversion [16], and PII [7]. Furthermore, we provide more qualitative results for class neurons (Figs. 11 to 18), intermediate neurons (Figs. 20 and 21), and results on disentangles polysemantic neurons (Fig. 22). Additionally, we extend our results with the performance of intermediate neurons (see Sec. 2.1), analysis of LRP on class neurons (see Sec. 2.2), concept-level visualization (see Sec. 2.3), analysis on predictions (see Sec. 2.4), scalability across differ-

ent architectures (see Sec. 2.5), and of the failure cases (see Sec. 2.6).

2.1. Performance on Intermediate Neurons

It is essential for us to quantify the performance of intermediate neuron visualization, and AUC (Area Under the Curve) and MAD (Mean Activation Difference), as proposed in [10], serve as valuable metrics for this purpose. In essence:

- **AUC** measures how well a neuron’s activation distinguishes between relevant and irrelevant stimuli by computing the area under the Receiver Operating Charac-



Figure 2. The effect of the transparency map in the VITAL framework. The **first row** represents the visualization *without* a transparency map and the **second row** represents the visualization *with* a transparency map.

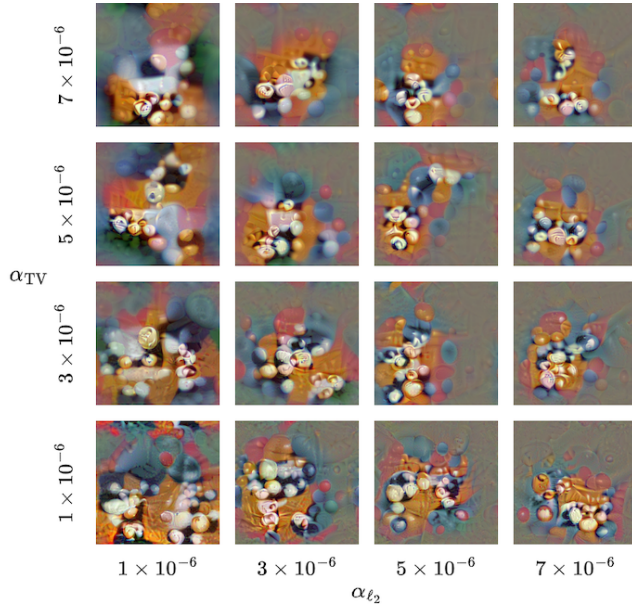


Figure 3. The effects of the parameters total variation α_{TV} and ℓ_2 norm α_{ℓ_2} on the final visualization in the VITAL framework.

teristic (ROC) curve, which plots the true positive rate against the false positive rate at various threshold settings. A higher AUC indicates that a neuron more effectively captures the intended concept, demonstrating a stronger alignment between its activations and the target representation.

- **MAD** quantifies the difference between the mean activation of the neuron on synthetic images and the mean activation on control data points. A higher MAD suggests that the neuron responds more strongly to synthetic images compared to real ones, indicating that the synthetic stimuli successfully elicit the neuron’s preferred feature representations.

Both metrics are essential for evaluating neuron visualization: AUC assesses a neuron’s discriminative power, deter-

block1	block2	block3	block4	block5	result
✓	✗	✗	✗	✗	
✗	✓	✗	✗	✗	
✗	✗	✓	✗	✗	
✗	✗	✗	✓	✗	
✗	✗	✗	✗	✓	
✓	✗	✗	✗	✓	
✓	✓	✓	✓	✓	

Figure 4. The effects of individual model components and their combination on the final visualization in the VITAL framework. Each block in the columns refers to a network component of ResNet50 (e.g., block1:conv1, block5:layer4).

mining how selectively it activates for a given concept, while MAD measures how strongly a neuron responds to synthetic stimuli relative to real ones, capturing the effectiveness of the visualization method. In Tab. 2, we present the average results for AUC and MAD across 90 neurons. We follow the experimental setup of [10], using a control dataset composed of the top-50 real ImageNet images that most strongly activate the target neurons, while the synthetic datasets are generated with three different seeds per neuron. The results demonstrate the superiority of VITAL over traditional feature visualization methods.

2.2. Analysis of Relevance on Class Neurons

For class neurons, we also experimented with incorporating relevance scores to factor out irrelevant activations. As

Method	Setup	AUC (\uparrow)	MAD (\uparrow)
Fourier	res: 224	0.3073	-0.8120
MACO	res: 224	0.2561	-0.9678
VITAL		0.5556	0.1587

Table 2. Comparison of methods on ResNet50 trained on Imagenet for intermediate neuron visualization through AUC and MAD metrics, indication the **best**.

for intermediate neurons, we used LRP and Guided Backpropagation to obtain the relevance maps of each building block for measuring their contribution to the final predicted class c for the given N images. In Fig. 5, we show a comparison between visualizing class neurons with and without relevance. As with intermediate neurons, incorporating feature relevance scores into activations and aligning their distributions would encourage background features in the FV to disappear. When comparing the visualizations of "agaric" with and without relevance, it is evident that the model focuses *only* on the mushroom's cap and its spore print color for classification. For class neurons, which encapsulate an *entire object*, we as humans also consider *each part* of the object to understand it (e.g., stem of a mushroom). For class neurons, this hence involves a trade-off between enhancing human interpretability and maintaining *faithfulness* to the model's exact reasoning mechanism, corresponding to optimization with and without relevance scores. An interesting line of future work would be to consider LRP on self-supervised models, which usually learn more than just one distinguishing feature of an object. There, class visualization involving an attribution method would make the most sense.

2.3. Visualization of Concepts

As part of Mechanistic Interpretability, people are interested in finding concept-based explanations of model behavior. These concepts might be feature directions encoded through multiple neurons in a layer, which can be, for example, discovered by CRAFT [6]. In VITAL, we obtain these directions as well as the images that highly activate the concepts through CRAFT. To optimize for feature directions, we modify the initialization of relevances of target neurons in LRP to reflect the weights given by the feature direction. Specifically, for each image's feature map at the penultimate layer, we compute the cosine similarity with the concept direction vector. Then, we obtain the pixel location of the highest cosine similarity to assign the direction vector as the initial relevance score and apply LRP as in intermediate neuron visualization. Through this modification, VITAL can give *meaning* to these feature directions. We provide several examples for concept visualization in Figs. 23 to 26.

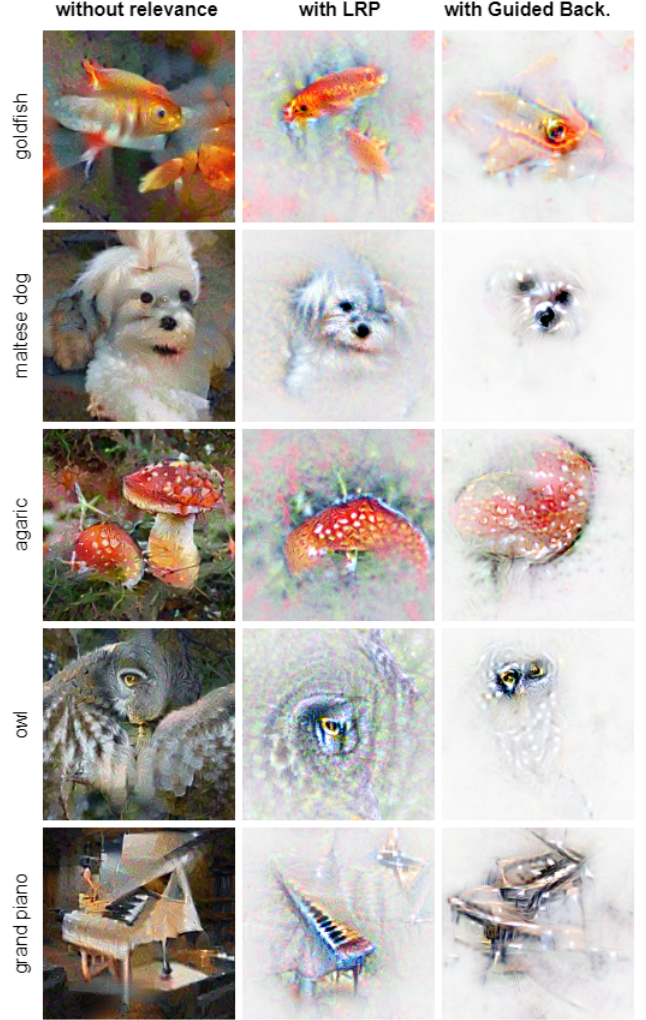


Figure 5. *Analysis on Relevance*. We performed an analysis to examine the impact of relevance information on the visualizations of class neurons on ResNet50. The findings highlight the effectiveness of LRP and Guided Backpropagation in finding the most significant regions for classification.

2.4. Analysis on Predictions

We further investigate the predictions of images produced from different methods on ResNet50 and observed that other methods including MACO seem to produce irrelevant features that mislead the model, producing predictions unrelated to the original class. We give two examples of prediction including the second-highest class score in Tab. 3.

2.5. Scalability Across Different Architectures

We demonstrate the scalability of VITAL by evaluating its performance across various architectures and conducting qualitative assessments (see Figs. 13 to 18). Our results highlight the robustness of our approach, whereas other methods fail to achieve similar adaptability and consistency

Images	Tiger		Maltese dog	
	#1 score	#2 score	#1 score	#2 score
ImageNet	Tiger 0.8625	Tiger cat 0.1363	Maltese dog 0.9757	Lhasa 0.0149
MACO	Tiger 0.5752	Apiary 0.1616	Silky terrier 0.4127	Coral reef 0.2002
DeepInv	Tiger 0.9984	Tiger cat 0.0013	Maltese dog 0.9989	Lhasa 0.0004
VITAL	Tiger 0.8609	Tiger cat 0.1382	Maltese dog 0.9797	Lhasa 0.0112

Table 3. Comparison of top-2 softmax scores of example classes across methods applied on ResNet50 trained on ImageNet. We use MACO with resolution 224 and DeepInv with batch size 64 (best setting in previous experiments) and indicate **misclassification**.

across different network designs. Furthermore, t-SNE projections in Fig. 9 reveal a similar trend in the embedding space. VITAL is the only method that reliably position generated features at the center of their respective clusters, capturing distinct characteristics that are recognizable.

2.6. Hardness Analysis and Failure Cases

While VITAL generally produces clearer and more conceptually relevant images, there are still cases where visualization quality suffers. These negative examples highlight areas for further refinement of our framework. In Fig. 19 we investigate these cases, including hardness of interpretation analysis of the generated visualizations with the help of the aforementioned user studies. We also offer analysis based on our interpretations. We observe it is harder for people to interpret that includes particularly complex scenes, such as the *vacuum cleaner* and *ambulance* from ResNet50, where the concept is less distinct. Additionally, due to the distribution-matching loss, local details are lost, leading to unrealistic structure in generated images—most notably seen in the *Persian cat* visualization from ResNet50 and the *husky* from ViT-L-32. Holistic user studies further indicate low confidence in intermediate neuron visualizations of ResNet50 (see Fig. 19), suggesting room for improvement in this area. Moreover, ViT-L-32 exhibits cases where certain classes, such as *scorpion* and *water snake*, are not clearly represented at all. These observations emphasize the need for a more comprehensive study of ViTs, considering the effects of individual blocks, regularization strategies, and transformation processes.

3. Ablation Studies

3.1. Effects of the Building Blocks on Visualization

We performed an analysis of how different components of a model affect the final visualization on ResNet50. As illustrated in Fig. 4, when we match the feature distribution

	Setup	Acc.	FID (\downarrow)	Zero-Shot Prediction	
		Top1 (\uparrow)		Top1 (\uparrow)	Top5 (\uparrow)
Vary $ X_{ref} $	rand5	99.47	40.30	57.60	85.87
	rand10	99.83	48.94	62.37	89.40
	rand20	99.87	54.45	62.90	90.20
	rand50	99.90	58.79	66.62	92.56
	rand100	99.77	59.28	65.67	90.50

Table 4. Comparison of class-specific sampling size for the reference images on ResNet50.

of the coarser layers, the resulting visualization primarily captures low-level information such as colors and textures, which we refer to as the style information. As we go deeper into the network, the visualizations progressively incorporate more contextual information such as the shape or the structure of an object at the cost of increased high-frequency noise. Accordingly, we observed that we can achieve a more realistic and proper visualization result by employing all the building blocks of our model that enable us to transfer the style into the context.

3.2. Effect of the Regularization Losses

In Fig. 3, we examine the impact of the parameters, α_{TV} and α_{ℓ_2} , of the auxiliary regularization loss that are used to further reduce noise and small artifacts in the generated image for intermediate neuron visualization on ResNet50. It should be noted that, in Fig. 3, we visualized the images without a transparency map to visualize the full extent of the impact of the regularization losses.

3.3. Effect of the Transparency Map

Irrelevant areas of the generated image stay mostly unchanged during the optimization, essentially representing noise. Analogous to Fel et al. [5], we suggest using transparency maps based on the importance of the image location during optimization to show relevant image parts only. In brief, we accumulate the gradients of our loss across each step in the optimization. As done in SmoothGrad [15], we average those gradients through the whole optimization process. We thus ensure the identification of the areas that have been most attended to by the network during the generation of the image. We illustrate the effect of the transparency map on ResNet50 in Fig. 2.

3.4. Effect of the Reference Images

For ResNet50, we systematically varied the X_{ref} size for randomly selected images from a given class (see Tab. 4 and Fig. 7), observing that VITAL remains robust and achieves saturation around 50 samples. This suggests that our method effectively captures the underlying feature distributions with a relatively small reference set. However, according to our preliminary analysis, when selecting random samples with-

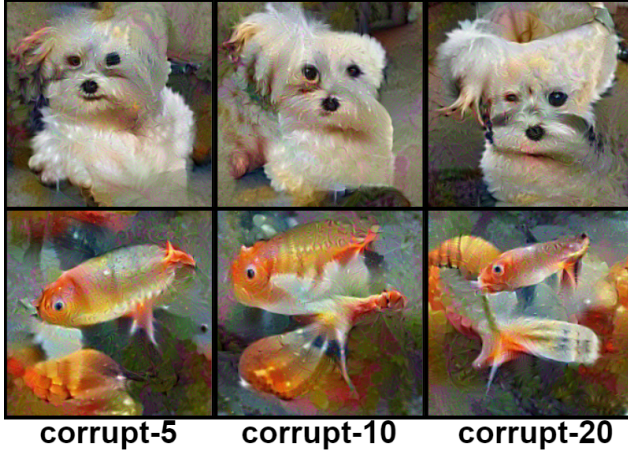


Figure 6. The qualitative impact of corrupting the reference images on class neuron visualization.

	Acc.	FID	Zero-Shot Prediction	
	Top1 (↑)	(↓)	Top1 (↑)	Top5 (↑)
Test Set	99.27	65.93	61.63	89.30
Train Set	99.90	58.79	66.62	92.56
Corrupt-5	99.97	72.73	61.03	88.80
Corrupt-10	99.83	86.73	56.20	85.50
Corrupt-20	99.93	113.30	47.30	77.40
CIFAR-10	-	-	85.47	99.09
Fourier	5.97	34.87	11.40	53.80
VITAL	100.00	0.55	78.30	98.90

Table 5. The quantitative impact of corrupting the reference images, test-set analysis and CIFAR-10 analysis on class neuron visualization with ResNet50.

out considering class alignment or activation guidance, the resulting visualizations lose coherence and fail to provide meaningful insights, highlighting the importance of structured sampling in our approach.

Additionally, we extended our embedding analysis to further validate the reference set strategy. As represented in Fig. 10, we clustered all training samples from three ImageNet classes into subgroups and generated representative VITAL visualizations per cluster using 50 nearest neighbor images per cluster. The resulting t-SNE plots show that VITAL images cover diverse intra-class modes without collapsing to a single mode, confirming that our approach preserves both local feature fidelity and global semantic diversity within the same class.

To verify that VITAL is not dependent on the training data and the dataset that is being used, we also consider test set examples for the reference and CIFAR-10 dataset, confirming that it can handle data beyond the original training data and the ImageNet dataset (Tab. 5). Finally, we performed a corruption experiment, in which we corrupted the reference set by gradually adding 5, 10, and 20 images from

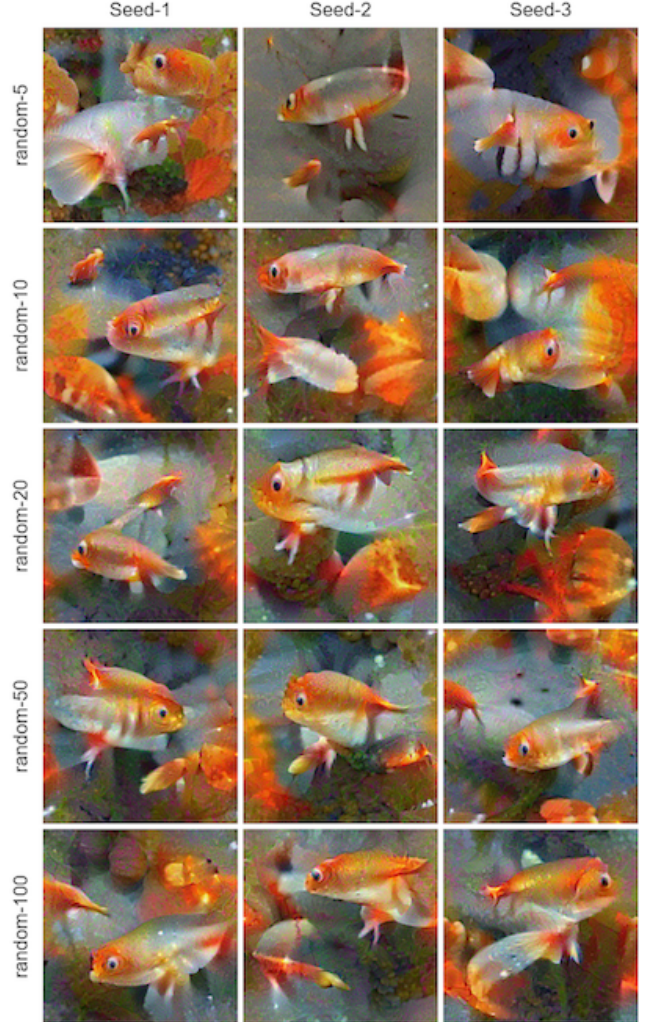


Figure 7. The effect of class-specific reference image sampling size on class neuron visualization.

outside the class (Fig. 6 and Tab. 5). As expected, image quality and metrics degrade progressively with increasing contamination; however, even with partial corruption, VITAL visualizations remain considerably more stable than prior FV methods.

3.5. Alternative Attribution Methods

We extend VITAL for class neuron, intermediate neuron and concept visualizations on ResNet50 by incorporating Guided Backpropagation as an alternative to LRP, providing additional insights into feature attributions and model interpretability. We provide qualitative results to compare Guided Backpropagation with LRP for class neurons in Fig. 5, intermediate neurons in Figs. 20 and 21, and concepts in Figs. 23 to 26.

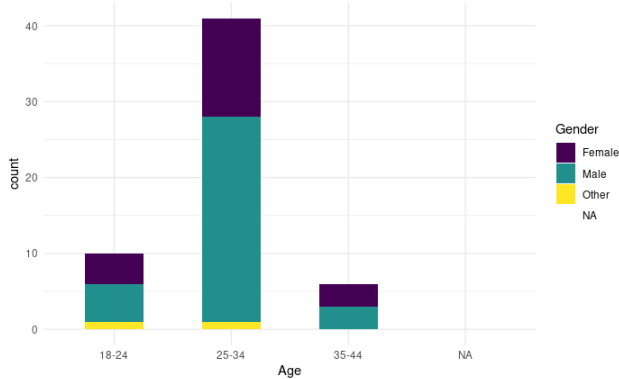


Figure 8. *Demographic Analysis.* We represent age and gender distribution of the participants in our user study.

4. Human Interpretability Study

In this section, we describe the details of our user study to quantitatively measure the performance of VITAL and different FV methods on *human interpretability*.

Participation. Participation in the study was voluntary, with 58 individuals taking part. Among these participants who disclosed their demographic information, 61.4% identified as male, 35.1% as female, while 3.5% selected the option "other". Regarding age distribution within a range, 17.5% aged 18-24, 71.9% aged 25-34, and 10.5% aged 35-44. We provide frequency distributions of the demographics with respect to age and gender in Fig. 8.

Study layout. The study design is described in the main paper, we will here describe the layout of the three parts of the study. We conducted the user study in Google Forms. Participants were initially redirected to a welcome page, where the study’s general purpose and procedures were clearly explained (see Fig. 27). Subsequently, they were presented with the first section of our user study, where given a single word, they evaluated how well a FV reflects the provided word. This section contains 10 sets of words and FVs in total numbered Q1-Q10 with a simple scoring system from 1 (worst) to 5 (best) to rank the visualizations (see Fig. 28). In the second section of our user study, users evaluated how well the FVs for an inner neuron reflect the provided reference images (highly activating on the target neuron). This section contains 10 sets of reference images in total numbered Q1-Q10 with a simple scoring system from 1 (worst) to 5 (best) to rank the visualizations (see Fig. 29). For section 3, participants were first asked to select one of three subsets (see Fig. 30), with each subset consisting of 9 questions from Q1-Q9 that required them to describe a given generated image with a word or a short description (see Fig. 31). To ensure comparability of methods, in each question corresponding to a given target word, each subset had one specific methods’ visualization for that question. For example,

in Q1, the target class was Espresso, and subset 1 had a FV of VITAL, subset 2 a FV of DeepInversion, and subset 3 a FV of MACO. Finally, participants were presented with an optional section on demographic analysis (see Fig. 32) before submitting the user study.

Analysis and Results. In Figs. 33 to 35, we provide a fine-grained analysis of each question across all sections of our user study complementing the results in the main paper. In particular, we provide class- or concept-specific score distribution for each method. We observe that, as before, our method performs favorably across all three tasks compared to other methods for each question. Furthermore, we see that VITAL yields consistently good results, showing better results in almost all cases across all study sections. There are specific classes, such as specific animals, or "hamburger" and "grand piano", where our method yields much more interpretable visualizations.

Holistic User Evaluation As a complement to our proposed user study, we conducted a validated user evaluation following the protocol from [5, 17]. In this study, we recruited $N = 42$ participants and replicated the setup using four randomly selected class neurons and nine randomly selected inner neurons from ResNet50. The user study is composed of two sections, where in section 1, we incorporated four different class neurons with 4 subset of questions, and in section 2, we incorporated 9 different intermediate neurons with 3 subset of questions. We have included a demonstration section to enhance clarity of the study. We provide layout snapshots from the user study in Fig. 36. We measured correctness based on participant confidence (maximum score of 3). For class neurons, VITAL achieved 100% correctness (2.81), outperforming MACO at 92.86% (2.51), Fourier at 90.48% (2.45), and DeepInv at 100% (2.62). Similarly, for intermediate neurons, VITAL demonstrated superior performance with 95.24% correctness (2.43), compared to MACO at 88.89% (2.14) and Fourier at 87.30% (2.05). We provide fine grained analysis in Fig. 37 for both section-1 and section-2. These results confirm that VITAL outperforms existing feature visualization (FV) methods in supporting interpretability.

References

- [1] Prompt Engineering for ImageNet. https://github.com/openai/CLIP/blob/main/notebooks/Prompt_Engineering_for_ImageNet.ipynb. 2
- [2] Dreaming to Distill: Data-free Knowledge Transfer via DeepInversion. <https://github.com/NVlabs/DeepInversion/tree/master>. 1
- [3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *The IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009. 1
- [4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *The International Conference on Learning Representations (ICLR)*, 2021. 1
- [5] Thomas Fel, Thibaut Boissin, Victor Boutin, Agustin Picard, Paul Novello, Julien Colin, Drew Linsley, Tom Rousseau, Remi Cadene, Lore Goetschalckx, Laurent Gardes, and Thomas Serre. Unlocking feature visualization for deep network with magnitude constrained optimization. In *The Advances in Neural Information Processing Systems (NIPS)*, pages 37813–37826. Curran Associates, Inc., 2023. 1, 3, 6, 8, 32, 33
- [6] Thomas Fel, Agustin Picard, Louis Bethune, Thibaut Boissin, David Vigouroux, Julien Colin, Rémi Cadenc, and Thomas Serre. Craft: Concept recursive activation factorization for explainability. In *The IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 2711–2721, 2023. 2, 5
- [7] Amin Ghiasi, Hamid Kazemi, Steven Reich, Chen Zhu, Micah Goldblum, and Tom Goldstein. Plug-in inversion: Model-agnostic inversion for vision with data augmentations. In *The International Conference on Machine Learning (ICML)*, pages 7484–7512, 2022. 1, 3
- [8] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *The IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2015. 1
- [9] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *The IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017. 1
- [10] Laura Kopf, Philine Lou Bommer, Anna Hedström, Sebastian Lapuschkin, Marina MC Höhne, and Kirill Bykov. Cosy: Evaluating textual explanations of neurons. In *The Advanced Conference on Neural Information Processing Systems (NIPS)*, 2024. 3, 4
- [11] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *The IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 11966–11976, 2022. 1
- [12] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017. <https://distill.pub/2017/feature-visualization>. 1, 3
- [13] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 1
- [14] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *The International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021. 1
- [15] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *CoRR*, abs/1706.03825, 2017. 6
- [16] Hongxu Yin, Pavlo Molchanov, Jose M. Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deep-inversion. In *The IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 3
- [17] Roland S. Zimmermann, Judy Borowski, Robert Geirhos, Matthias Bethge, Thomas S. A. Wallis, and Wieland Brendel. How well do feature visualizations support causal understanding of cnn activations? *The Advances in Neural Information Processing Systems (NIPS)*, 34:24369–24381, 2021. 8, 32, 33

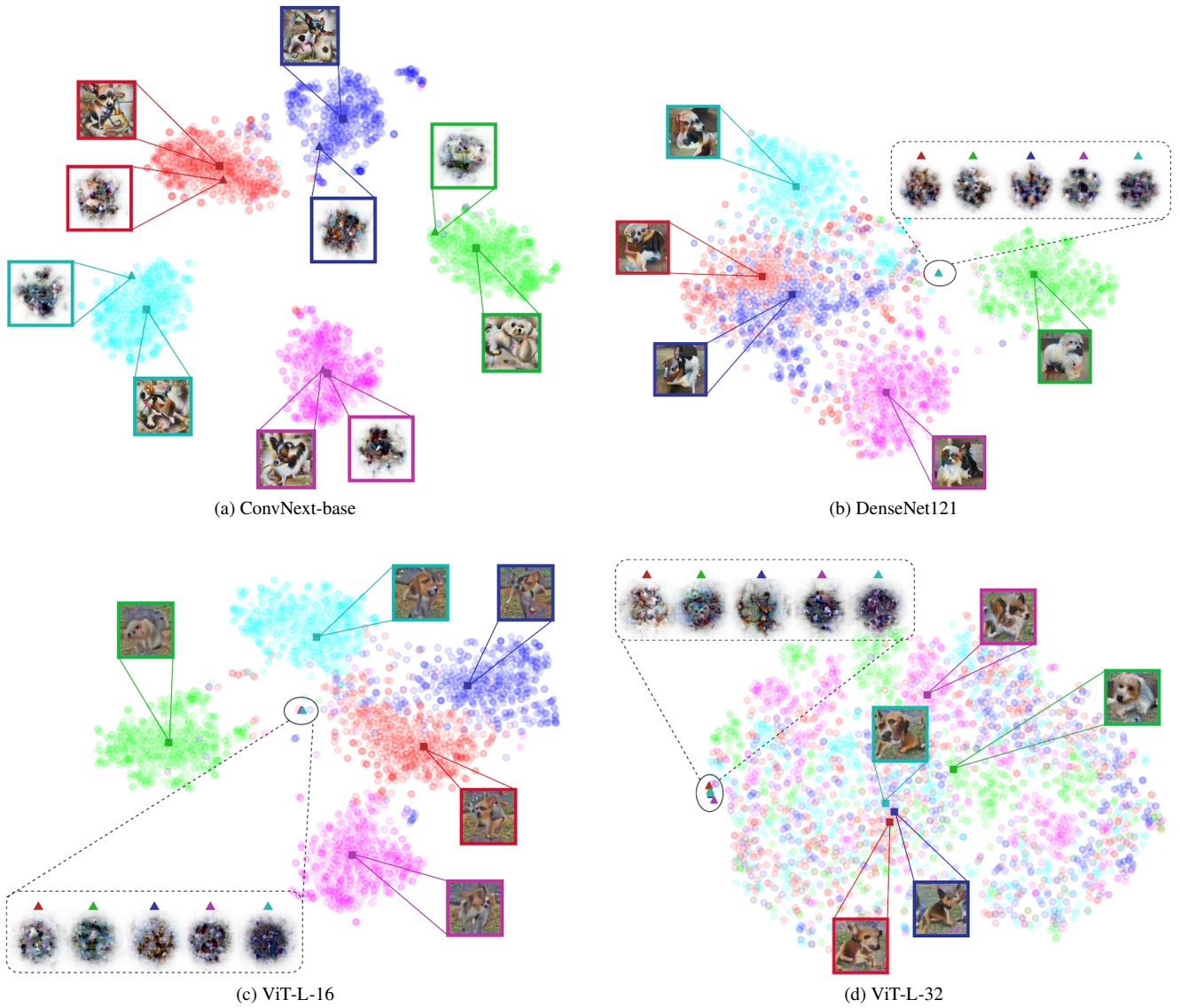


Figure 9. *t-SNE projection of embedding*. We show a low-dimensional tSNE embedding of the features at the penultimate layer for five dog breeds indicated by color across different architectures. Transparent circles are original training images and FVs are indicated by symbols: ■: VITAL, ▲: MACO.

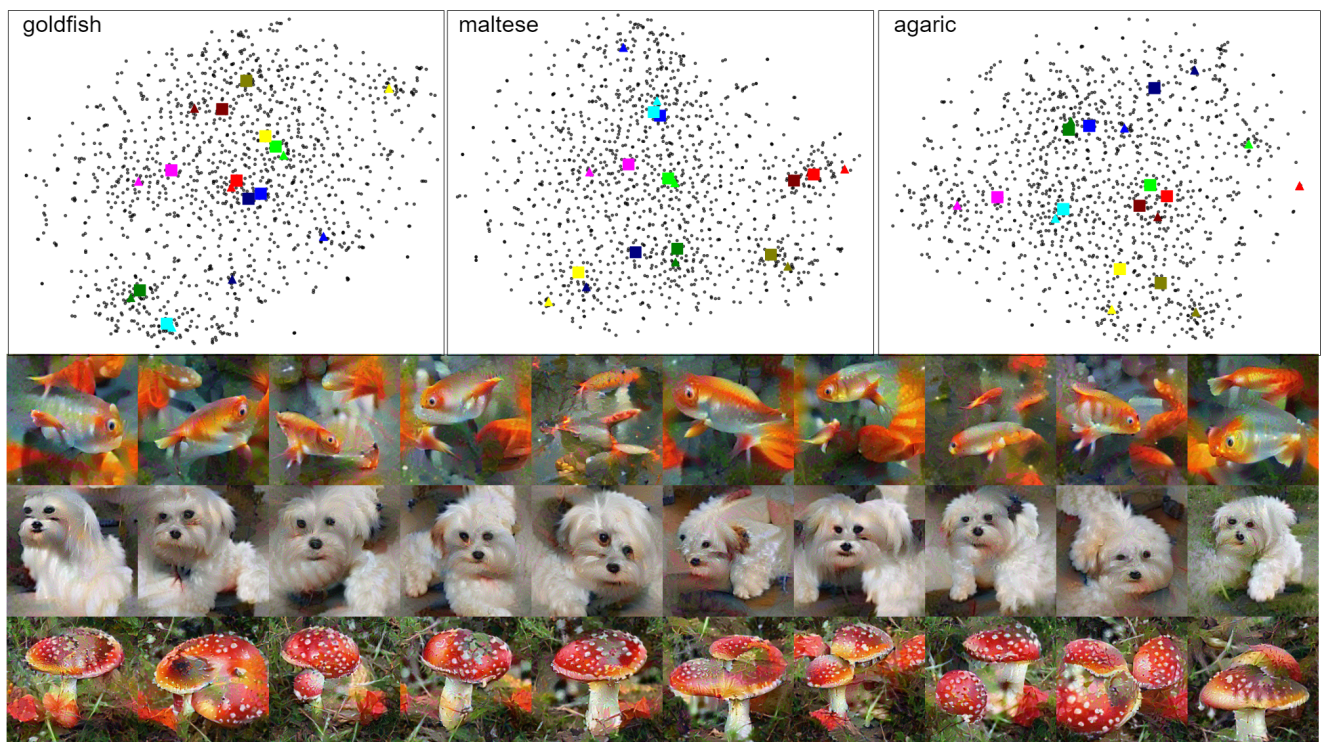


Figure 10. VITAL visualizations generated from clustered ImageNet training samples (3 classes shown) using 50 nearest-neighbor reference images per cluster (10 clusters). ■: VITAL, ▲: cluster center.

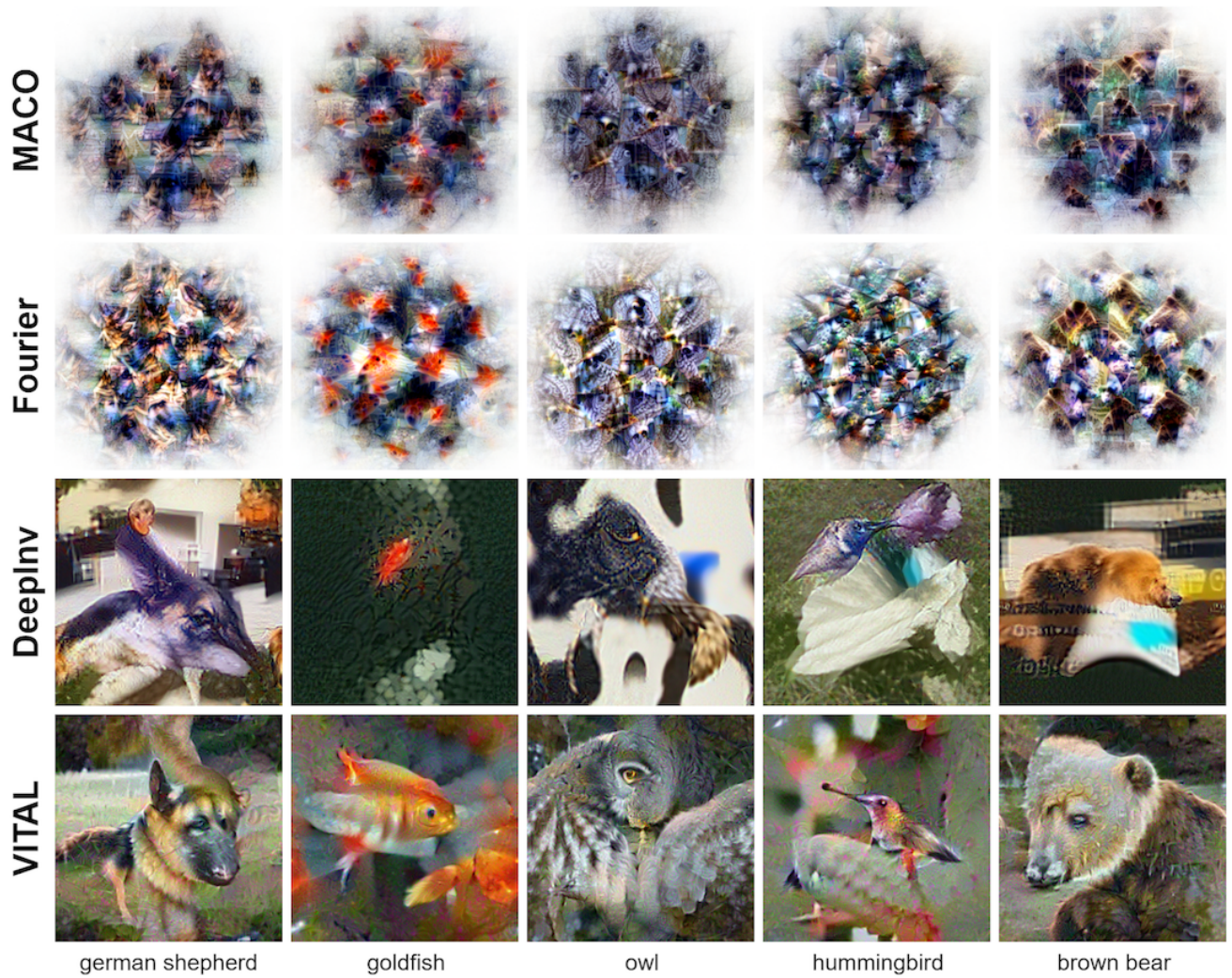


Figure 11. *Example class visualizations.* We provide more class visualizations for different classes (**columns**) of ImageNet for a trained ResNet50 model.

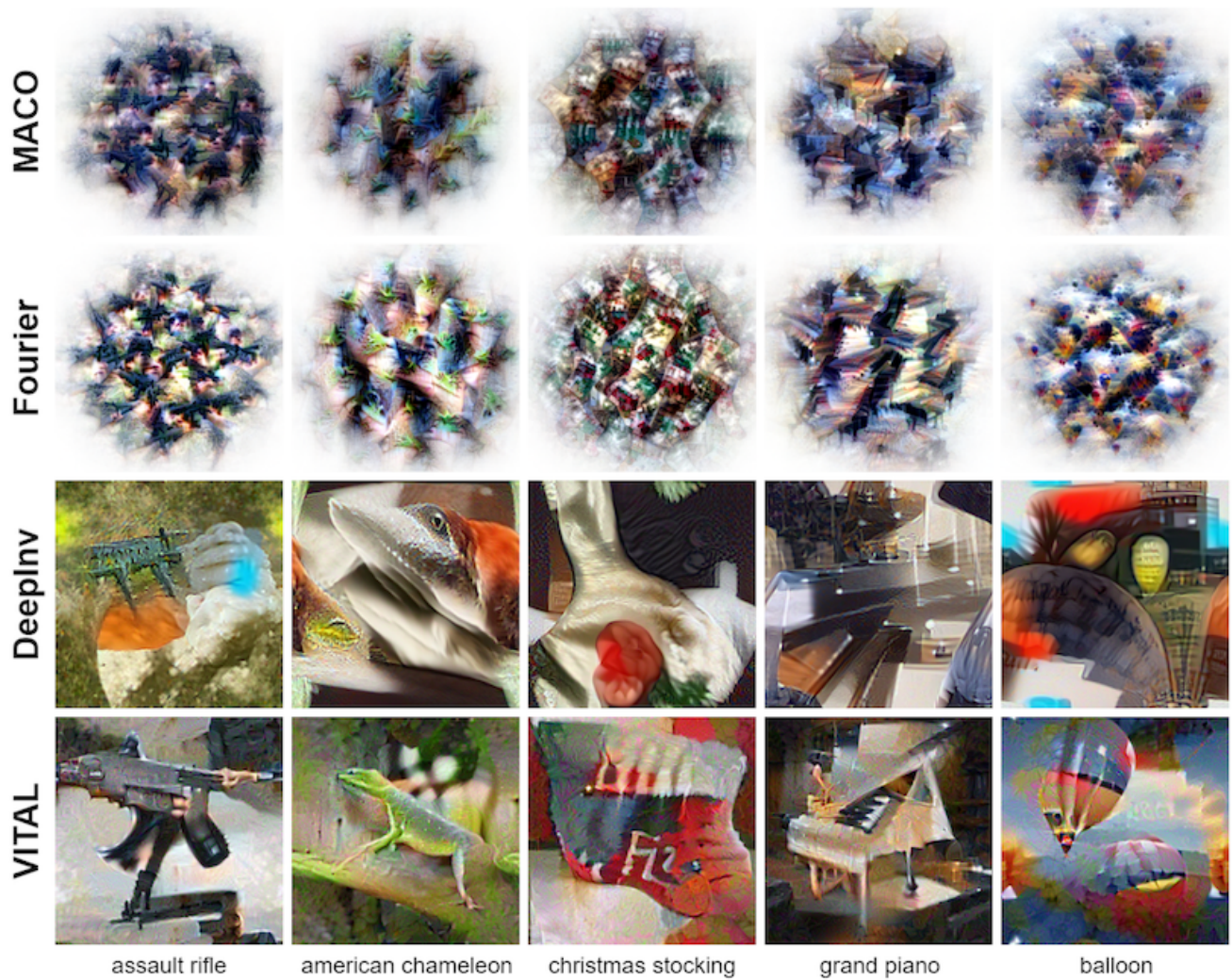


Figure 12. *Example class visualizations.* We provide more class visualizations for different classes (**columns**) of ImageNet for a trained ResNet50 model.

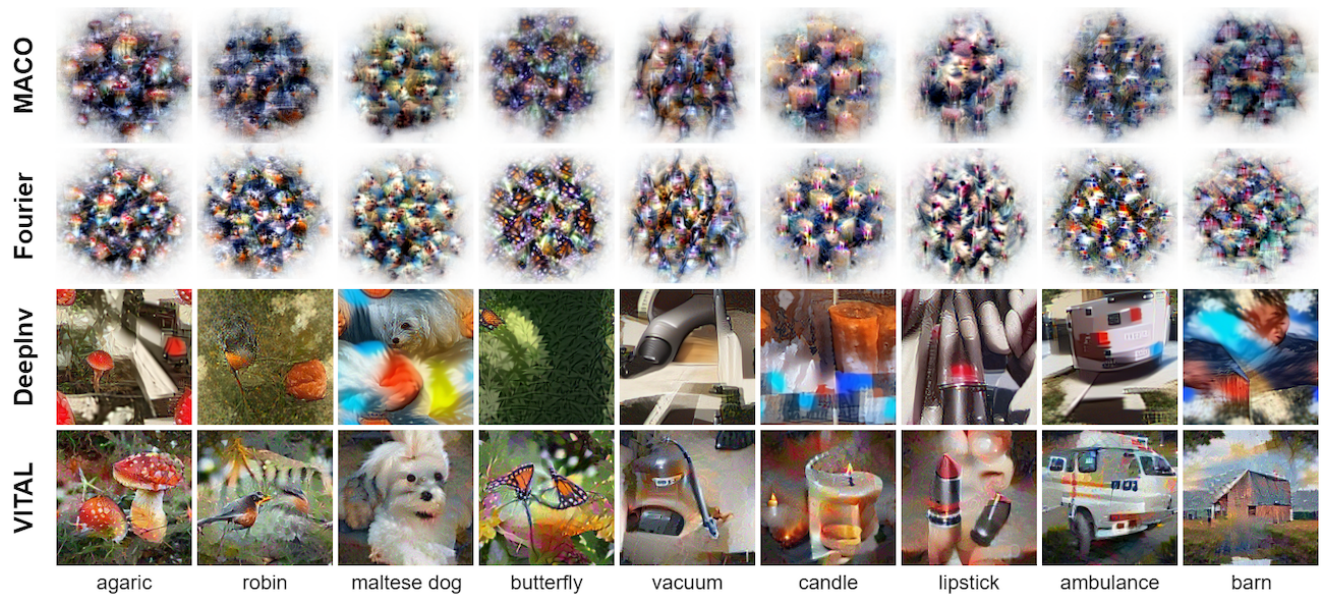


Figure 13. *Example class visualizations.* We provide class visualizations for different classes (columns) of ImageNet for a trained ResNet50 model. Existing work, in particular MACO and standard Fourier-based FV (top 2 rows), show highly repetitive patterns that are hard to understand. DeepInversion (3rd row) yields more understandable visualizations, yet suffers from artifacts that make it challenging to interpret. VITAL arguably yields much more interpretable and realistic visualizations, yet, as all methods, has problems with complex spatial arrangements (see the ambulance).



Figure 14. *Example class visualizations.* We provide more class visualizations of VITAL for different classes (**rows**) of ImageNet for different models (**columns**).

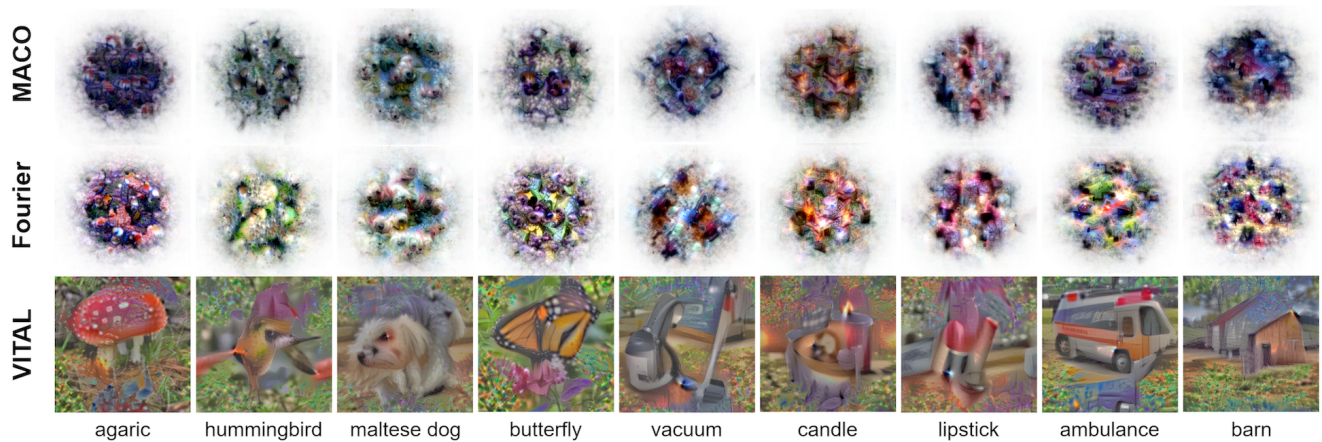


Figure 15. *Example class visualizations.* We provide more class visualizations for different classes (**columns**) of ImageNet for a trained ViT-L-16 model.



Figure 16. *Example class visualizations.* We provide more class visualizations for different classes (**columns**) of ImageNet for a trained ViT-L-32 model.

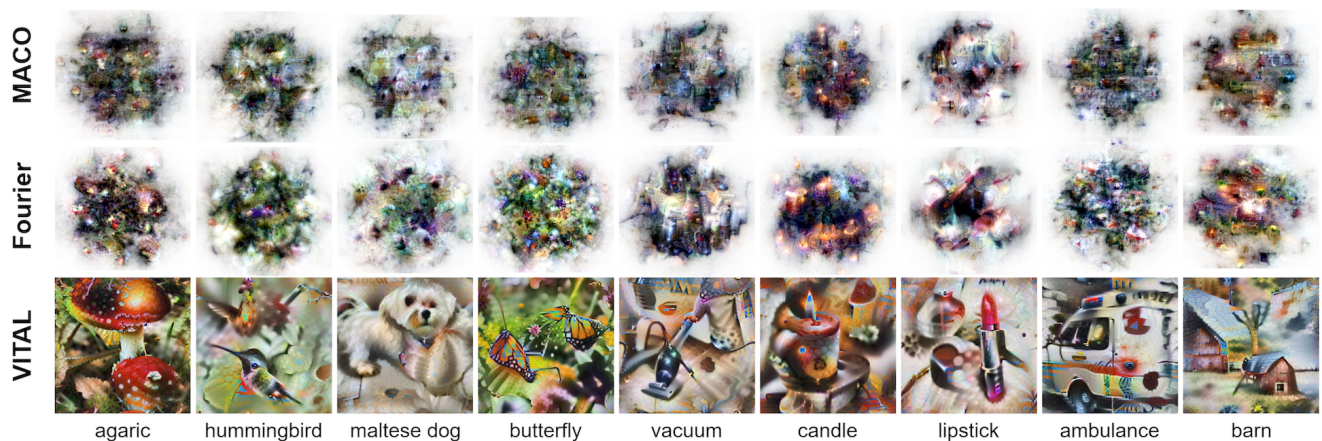


Figure 17. *Example class visualizations.* We provide more class visualizations for different classes (**columns**) of ImageNet for a trained ConvNext-base model.

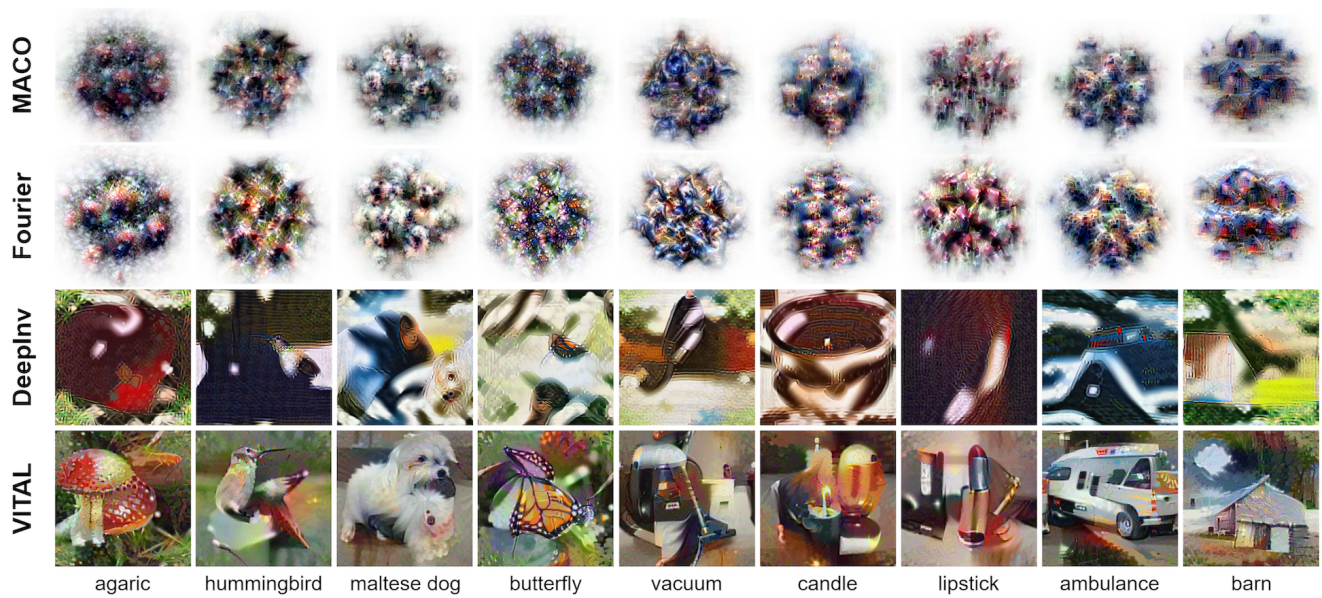


Figure 18. *Example class visualizations.* We provide more class visualizations for different classes (**columns**) of ImageNet for a trained DenseNet121 model.

class neurons



vacuum cleaner ambulance Persian cat

intermediate neurons



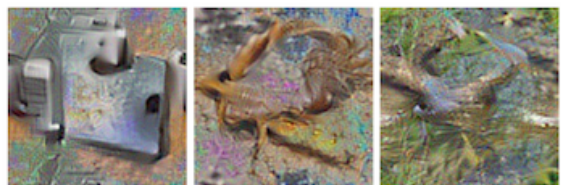
neuron #208 neuron #197 neuron #1232

(a) ResNet50

class neurons



vacuum cleaner husky lens cap



switch scorpion water snake

(b) ViT-L-32

Figure 19. The example failure cases in visualization quality for both ResNet50 and ViT-L-32.

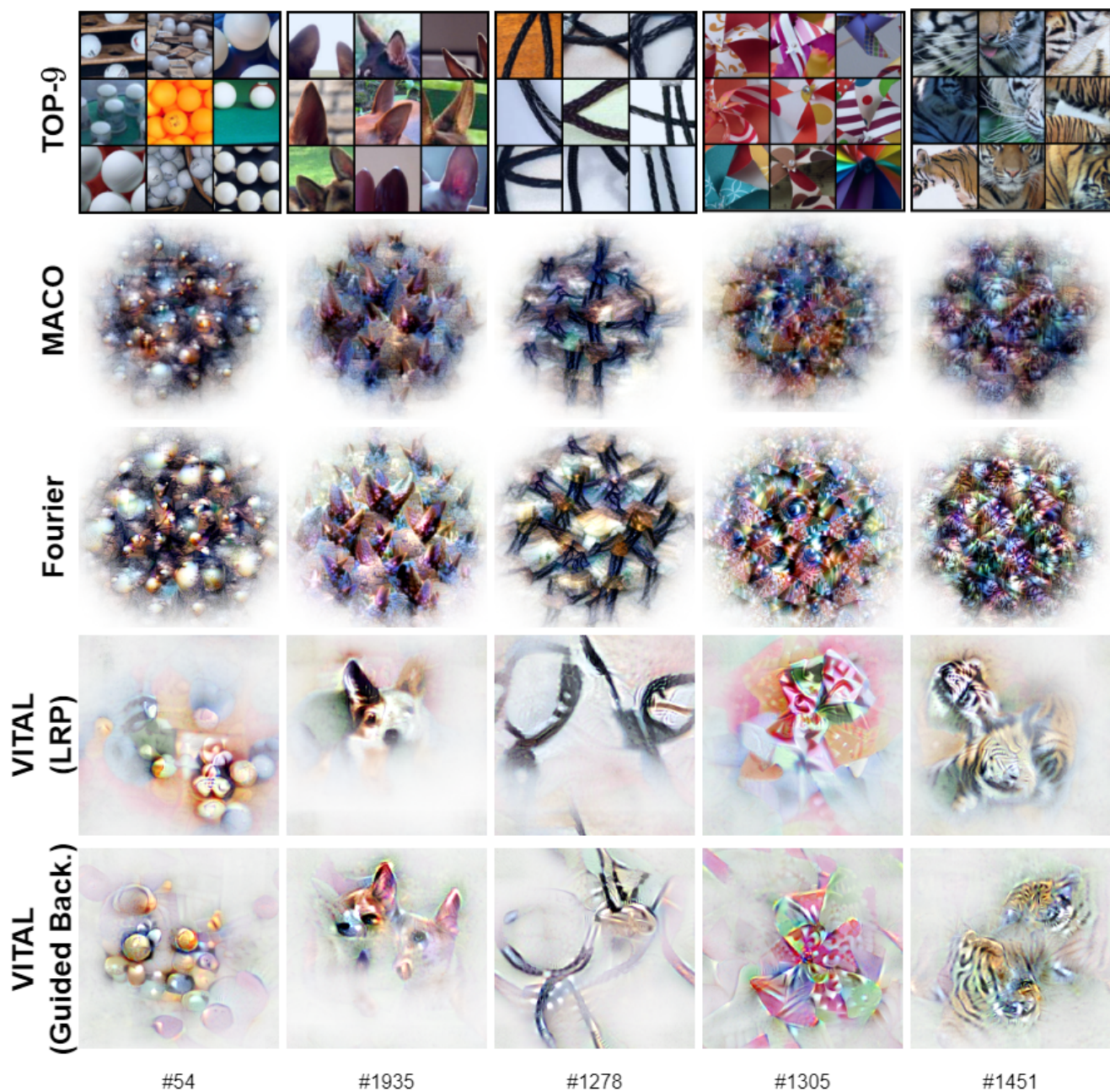


Figure 20. *Example intermediate neuron visualizations.* We provide visualizations for four randomly selected intermediate neurons (**columns**) of a trained ResNet50 model.

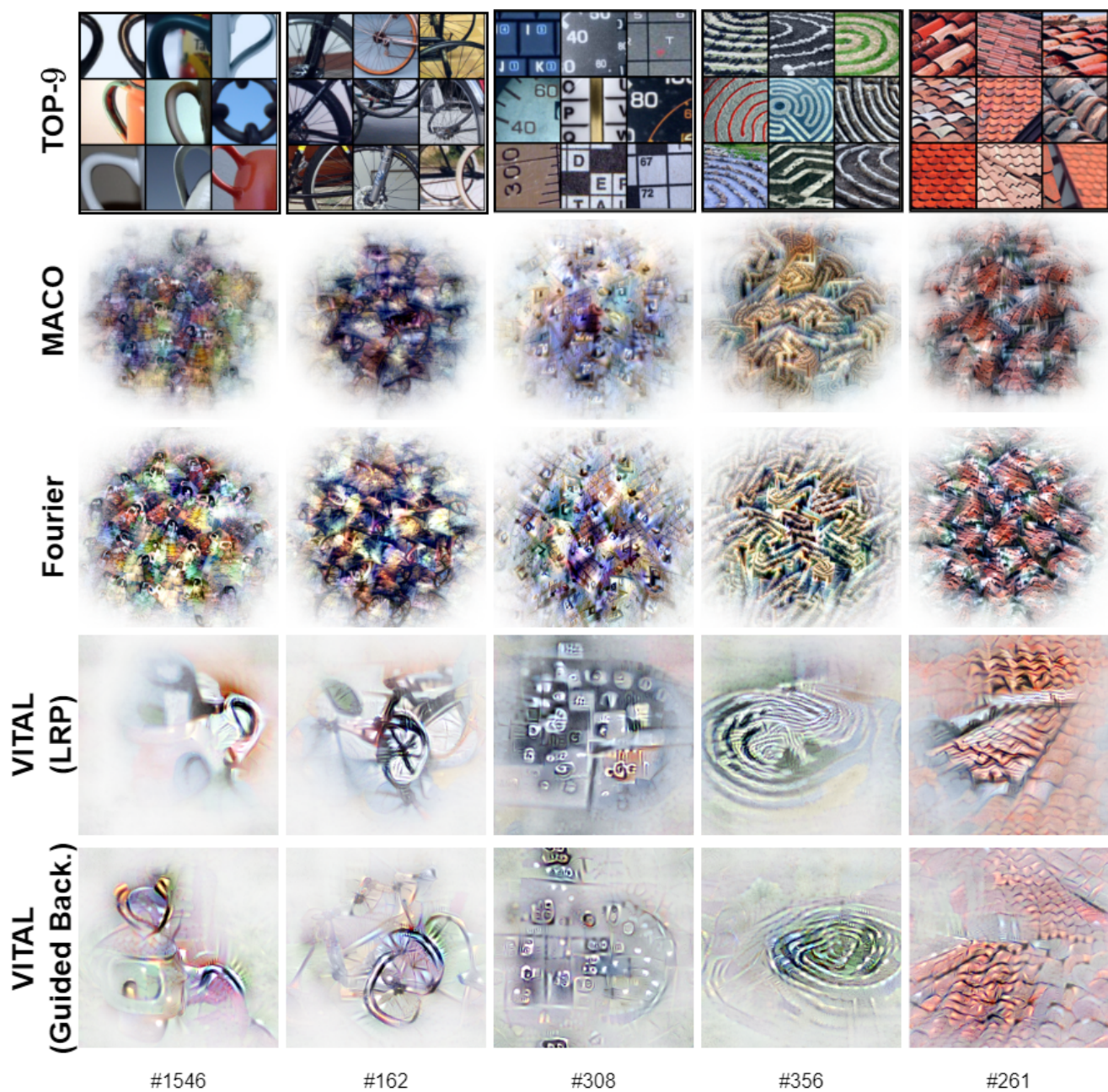


Figure 21. *Example intermediate neuron visualizations.* We provide visualizations for four randomly selected intermediate neurons (**columns**) of a trained ResNet50 model.



Figure 22. *Disentangling polysemanticity*. We provide four example visualizations from MACO with ResNet50 that generate visualizations that strongly activate for unrelated concepts. For each example, the first column represents the MACO visualization and the second represents the disentangled concepts from VITAL. Specifically, channel (**#485**) activates both on "burrito" and "dog body", channel (**#909**) activates both on "mattress" and "race car", channel (**#1524**) activates on "submarine", "lotion" and "bulb", channel (**#1431**) activates both on "abacus" and "bell pepper".

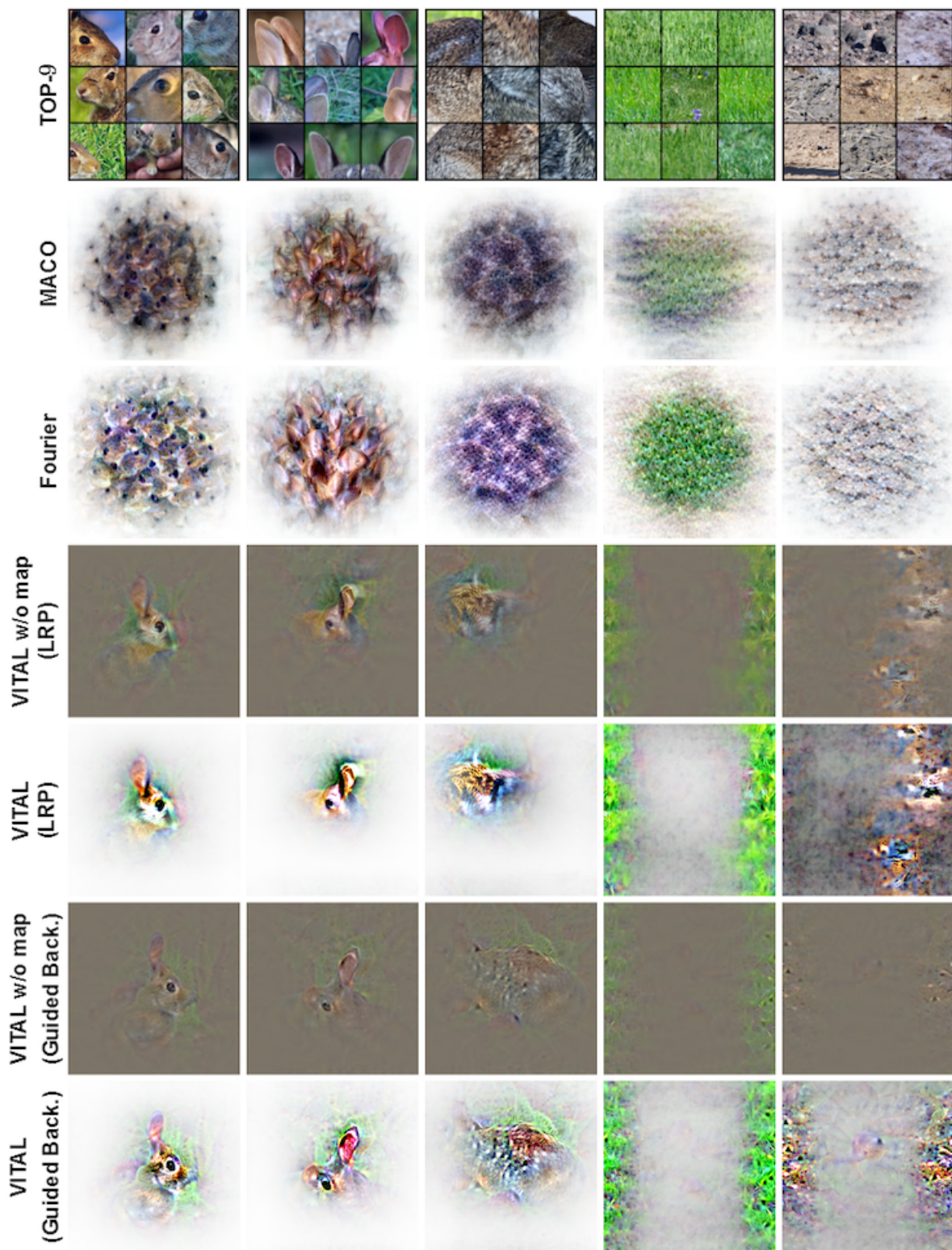


Figure 23. *Visualizing concepts*. We present example visualizations of the top five concepts identified using CRAFT for ResNet50. In this example, for the selected class **rabbit**, the top five concepts are identified as "rabbit face", "rabbit ear", "rabbit fur", "grass", and "surface".

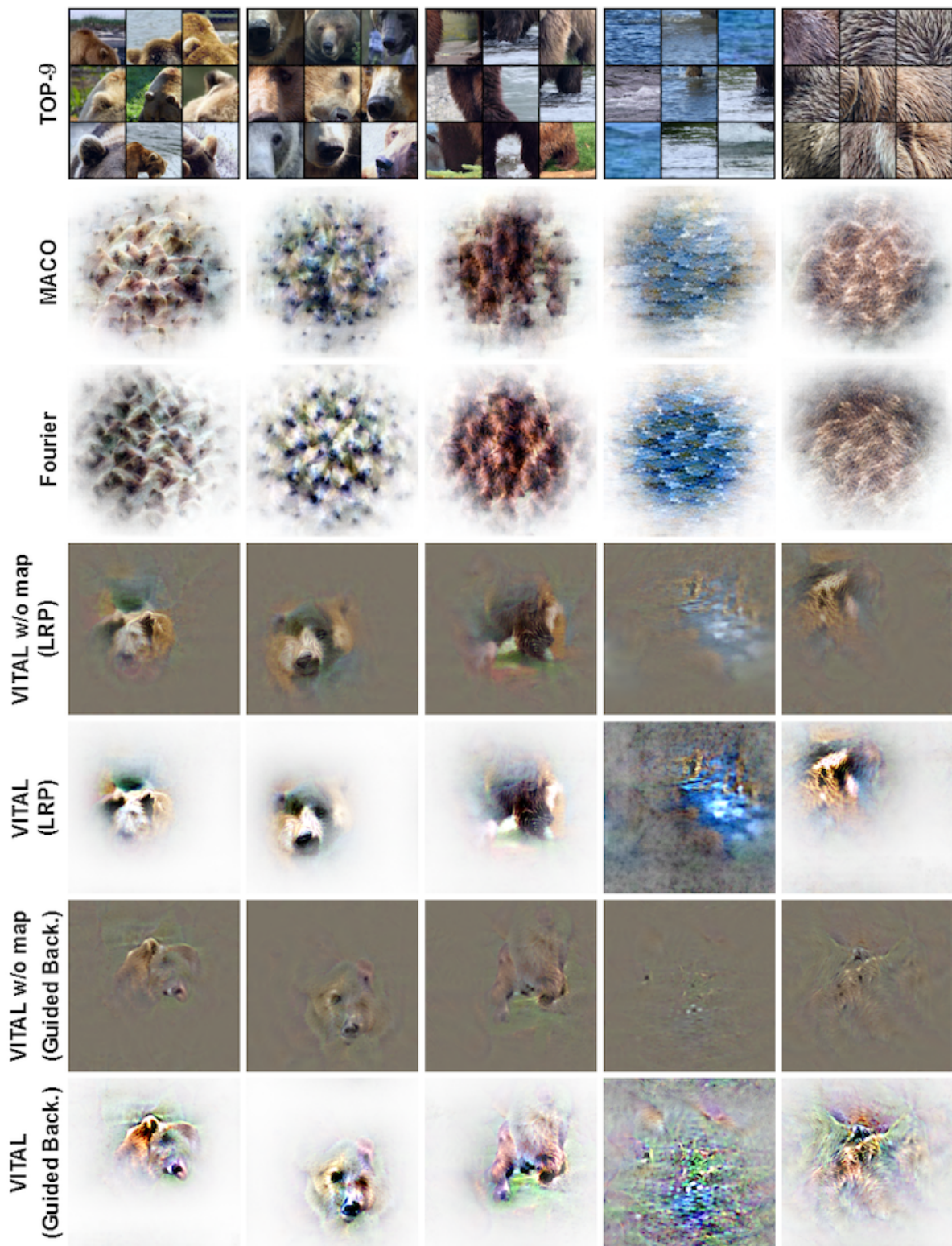


Figure 24. *Visualizing concepts*. We present example visualizations of the top five concepts identified using CRAFT for ResNet50. In this example, for the selected class **bear**, the top five concepts are identified as "bear ear", "bear face", "bear leg", "water", and "spiky fur".

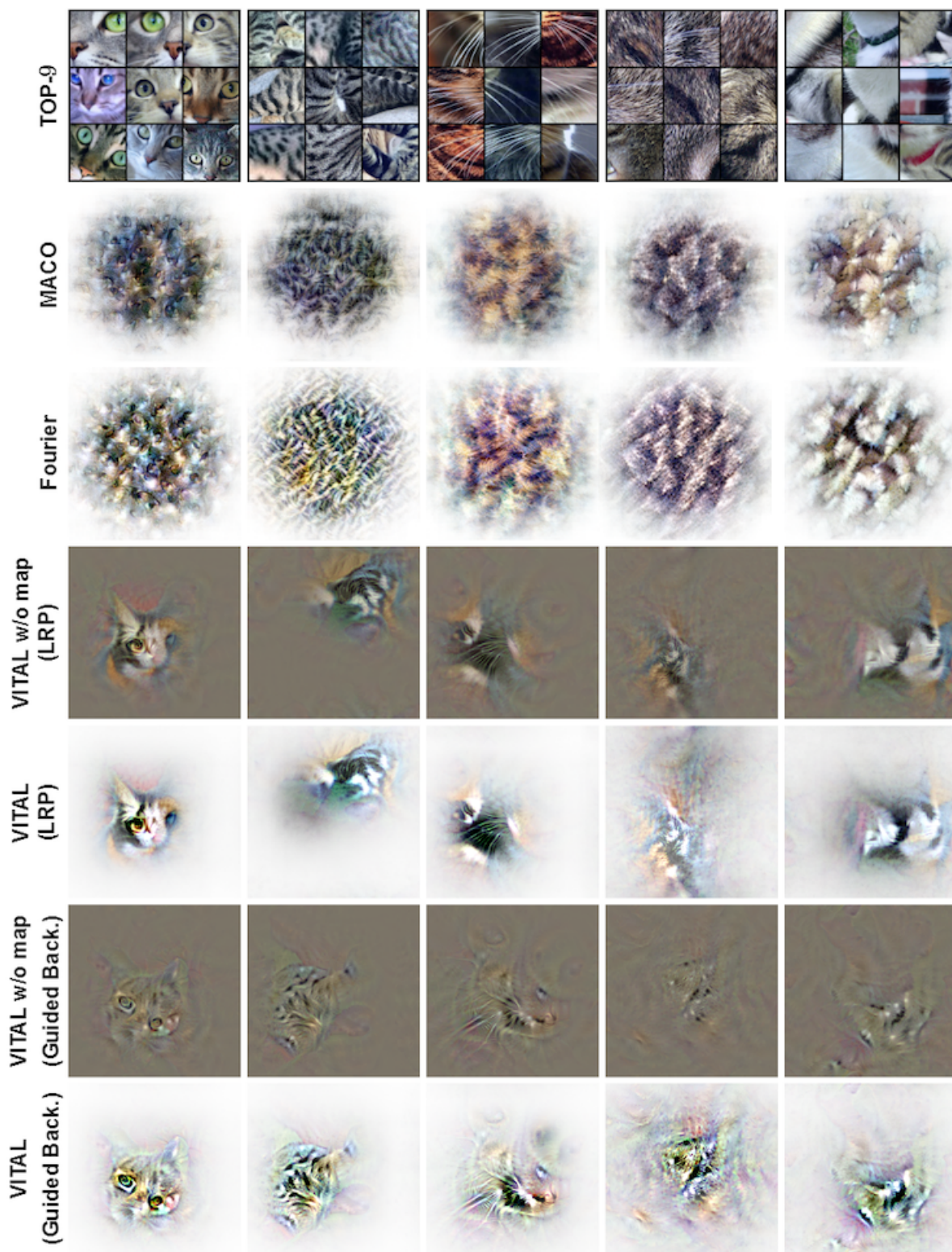


Figure 25. *Visualizing concepts.* We present example visualizations of the top five concepts identified using CRAFT for ResNet50. In this example, for the selected class **tabby cat**, the top five concepts are identified as "cat face", "fur with stripes," "cat whisker", "brown fur", and "white fur".

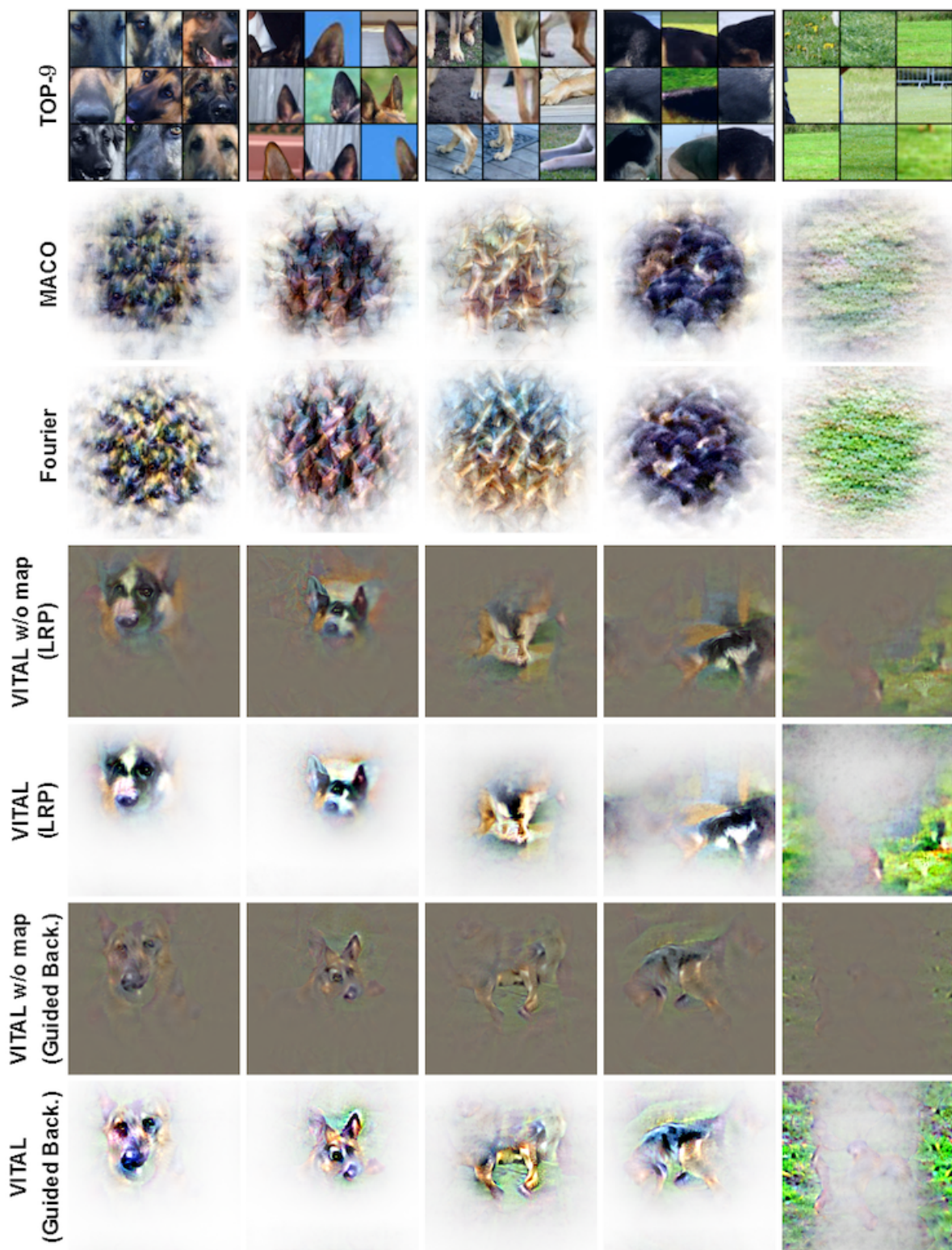


Figure 26. *Visualizing concepts.* We present example visualizations of the top five concepts identified using CRAFT for ResNet50. In this example, for the selected class **german shepherd**, the top five concepts are identified as "dog face", "dog ear", "dog leg", "dog body", and "grass".

User Study on Human Interpretability of Feature Visualizations

This study aims at the evaluation of methods that provide visualizations of neural network components, which help in understanding the reasoning process of neural networks. This survey is divided into three sections:

Section 1: Ranking Images Based on Word Alignment

You will be presented with questions each containing a **single word** and a **set of images**. Please rank the images on a scale of 1 to 5, where **1 is worst** and **5 is best**, considering how well the images are reflecting the provided word and how interpretable the images are to that extent.

Section 2: Ranking Visualizations Based on Similarity to Reference Images

In this section, you will be presented with multiple **reference images**. Alongside, you will see a set of visualizations. Please rank the visualizations based on how well they reflect the reference images and how interpretable are to that extent on a scale of 1 to 5, where **1 is worst** and **5 is best**.

Section 3: Describing Each Image in a Single Word

In this section, you will be presented with a **set of images**. Please write a **single word** that best describes the given images.

Answering should require less than 15 minutes and we highly appreciate your help. If you have any questions, please contact



Next

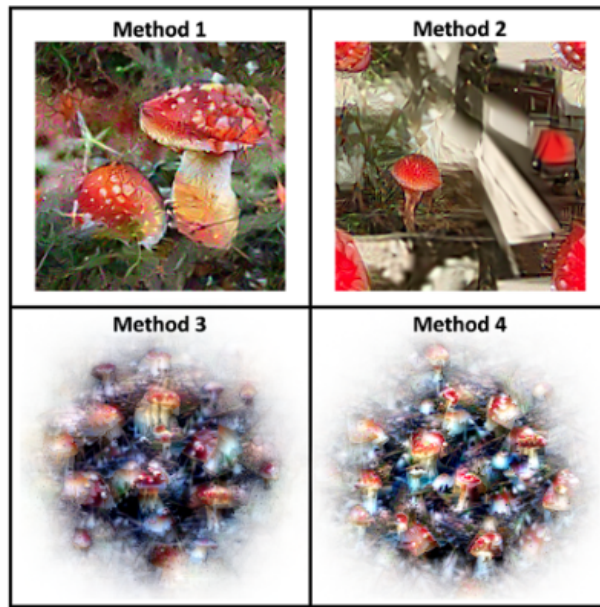
Clear form

Figure 27. *Welcome page.* A screenshot of the landing page of our user study.

Questions (Section 1)

This section contains the images you need to rank on a five-point scale, where 1 reflects the **worst** match and 5 reflects the **best** match based on how well the images reflect the provided word and how interpretable the images are to that extent. You can use the same scale for multiple methods. There are **10 sets of words** in total from Q1-Q10.

Q1. *
Word: agaric



	1	2	3	4	5
Method 1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Method 2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Method 3	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Method 4	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Figure 28. *Layout Section 1*. A screenshot that shows the content of section 1, including the task received with further instructions and a sample question.

Questions (Section 2)

This section contains the images you need to rank on a five-point scale, where 1 reflects the **worst** match and 5 reflects the **best** match based on how well the images reflect the provided reference images and how interpretable the images are to that extent. You can use the same scale for multiple methods. There are **10 sets of reference images** in total from Q1-Q10.

Q1. *




	Reference Images		Method 1		Method 2	
						
		1	2	3	4	5
Method 1		<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Method 2		<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Figure 29. *Layout Section 2*. A screenshot that shows the content of section 2, including the task received with further instructions and a sample question.

Questions (Section 3)

Please choose a random option here (to select a random subset)

Subsets: *
☐ #
☐ \$
☐ %

BackNext

Clear form

Figure 30. *Layout Section 3 subset selection.* A screenshot of the page that requires the participants to select a seed from 3 different subsets that determines the questions of section 3.

Questions (Section 3)

This section contains the images you need to describe in a single word. There are **9 sets of images** in total from **Q1-Q9**.


Q1. *

Your answer _____

Figure 31. *Layout Section 3.* A screenshot that shows the content of section 3, including the task received with further instructions and a sample question.

[OPTIONAL] Demographic Questions

This section is entirely optional and is intended to understand the demographics of the respondents. Feel free to leave any or all of these questions blank if you prefer. Thank you for participating in this study!

Age range

☐ Under 18

☐ 18-24

☐ 25-34

☐ 35-44

☐ 45-54

☐ 55-64

☐ Over 65

Gender

☐ Male

☐ Female

☐ Other

Back

Submit

Clear form

Figure 32. *Layout demographic questions.* A screenshot that shows the (optional) questions on age and gender.

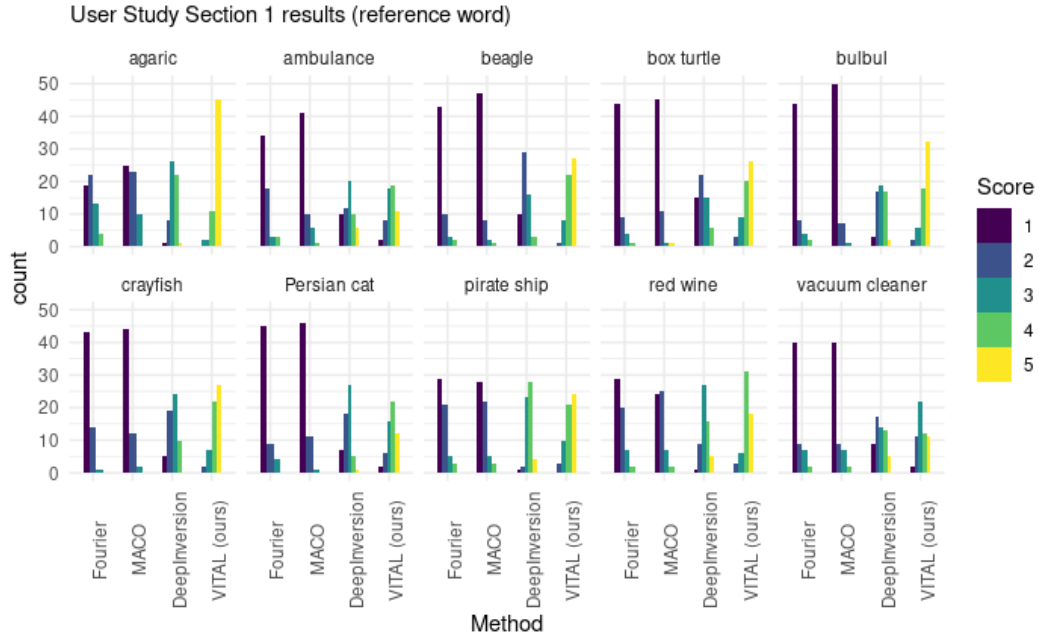


Figure 33. The statistics on the scores for the different methods obtained for the first part of our user study, separated by class.

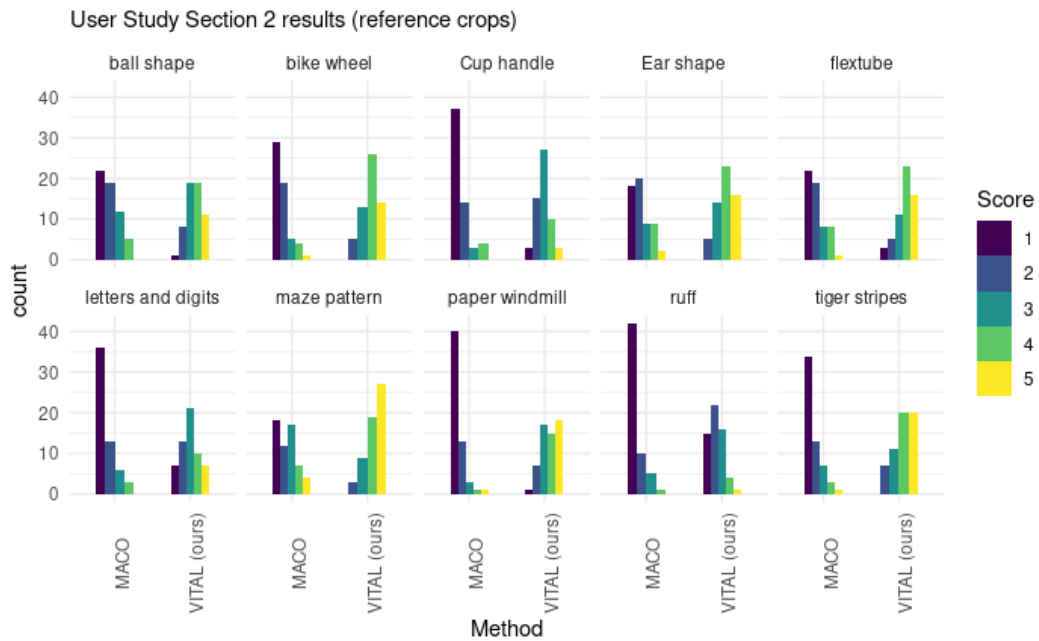


Figure 34. The statistics on the scores for the different methods obtained for the second part of our user study, separated by concept.

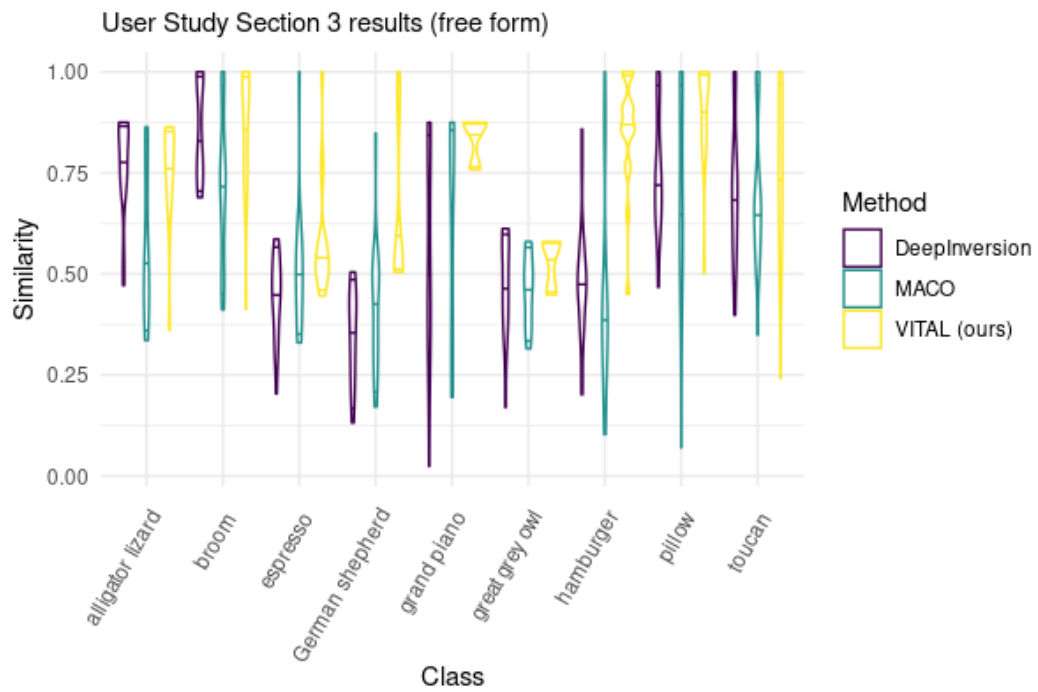


Figure 35. The violin plots with median, 5% and 95% quantiles of the achieved similarity for the last part of our user study, separated by class.

User Study on Human Interpretability of Feature Visualizations

This study focuses on **evaluating tools that create visual representations** of how parts of a neural network process images. These visualizations are meant to help people understand **how the network perceives information**. To test this, the study looks at whether these visual tools help users predict how changes to an image—like occlusions—affect the activity of specific parts of the network. By comparing different visualization methods, we explore **how useful these visualization tools are** for interpreting and understanding the behavior of neural networks.

In the next section, we provide a demonstration of how the survey test works, followed by two different sections that help us evaluate different visualization techniques. Answering should require less than 15 minutes and we highly appreciate your help. If you have any questions or concerns, please contact

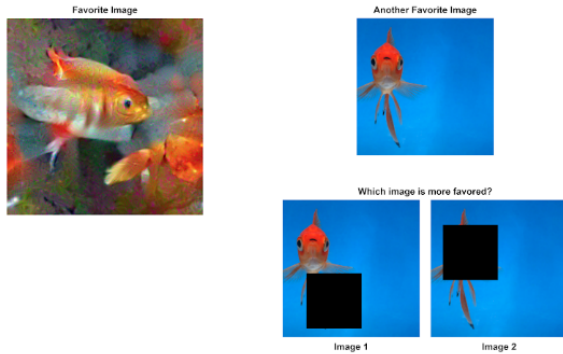
[Next](#)[Clear form](#)

(a) Welcome Page

Questions (Section 1)

Which image at the bottom right is **more** favored by the machine? In other words, in which image do you still see the visualized aspect of the favorite images? After selecting an image, rate your confidence from 1 (lowest) to 3 (highest). There are **4 questions** in total.

Q1: Which image does the machine *prefer*? *



Q1 Confidence *

low confidence 1 2 3 high confidence

☐ ☐ ☐

(b) Section-1

Questions (Section 2)

Which image at the bottom right is **more** favored by the machine? In other words, in which image do you still see the visualized aspect of the favorite images? After selecting an image, rate your confidence from 1 (lowest) to 3 (highest). There are **9 questions** in total.

Q1: Which image does the machine *prefer*? *



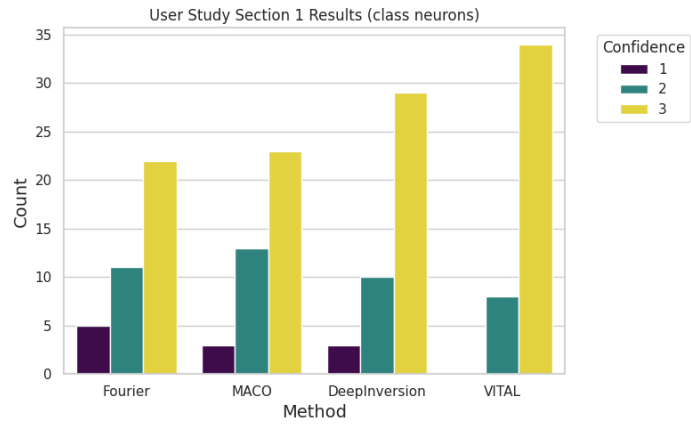
Q1 Confidence *

low confidence 1 2 3 high confidence

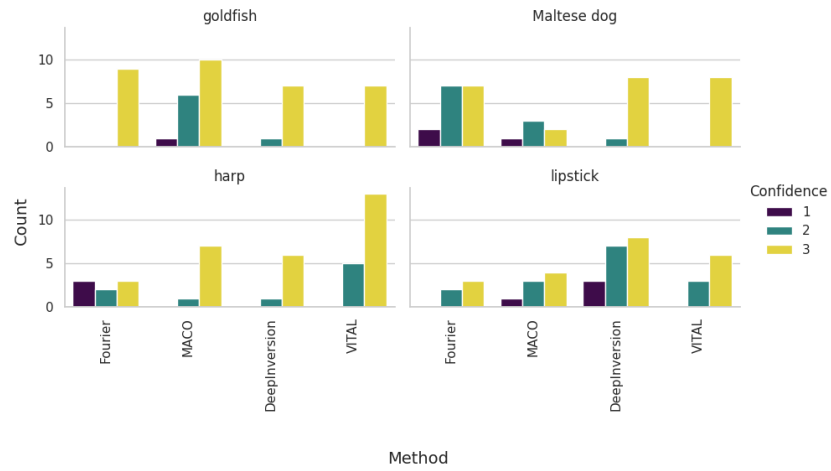
☐ ☐ ☐

(c) Section-2

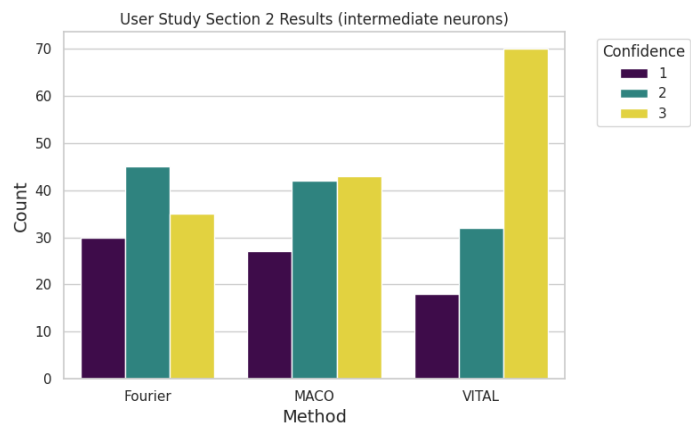
Figure 36. Layout of the validated user study from [5, 17], including the welcome page and example questions from section-1 and section-2.



(a) Section-1 Summary



(b) Section-1 Class-Specific



(c) Section-2 Summary

Figure 37. The statistics on the scores for the different methods obtained for the holistic user study [5, 17].